UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/675,429	09/30/2003	David A. Luick	ROC920030294US1	6193

7590 06/01/2006

Robert R. Williams
IBM Corporation, Dept. 917
3605 Highway 52 North
Rochester, MN 55901-7829

EXAMINER	
STOYNOV, STEFAN	

ART UNIT	PAPER NUMBER
2116	

DATE MAILED: 06/01/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/675,429	LUICK, DAVID A.	
	Examiner	Art Unit	
	Stefan Stoynov	2116	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *06 April 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-49* is/are pending in the application.

4a) Of the above claim(s) *28-30 and 38-40* is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-8,14,18-26,31-37,41,45 and 49* is/are rejected.

7)☒ Claim(s) *9-13,15-17,27,42-44 and 46-48* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *30 September 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

Election/Restrictions

Applicant's election without traverse of claims 1-27, 31-37, and 41-49 in the reply filed on 04/06/2006 is acknowledged.

Claims 28-30 and 38-40 were canceled.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

Claims 1-7 and 31-37, are rejected under 35 U.S.C. 103(a) as being unpatentable over Greene, U.S. Patent No. 6,775,787 in view of Gupta et al., U.S. Patent No. 5,996,083, and further in view of DeLano et al., U.S. Patent No. 5,337,415.

Regarding claim 1, Greene discloses a method for managing power dissipation in the processor core, the method comprising:

re-encoding an opcode of an instruction to incorporate a power token (column 4, lines 18-23, lines 35-50, column 5, lines 12-21);

adjusting the power dissipation in the processor core based upon a state of management control bits associated with the power dissipation, in response to a dynamic power count for the processor core based on issuance of the instructions (column 5, line 12 – column 6, line 15, column 7, line 31 – column 8, line 13, column 9 , lines 37-48, line 66 – column 10, line 15, lines 27-32).

Greene fails to disclose the power token comprising a bit to indicate a unit of the processor core to turn off during execution of the instruction.

Gupta teaches a power control register (FIG. 2, 108) including a plurality of fields for individually controlling the power consumption of the individual functional units within the microprocessor (column 5, lines 45-57, lines 61-67, FIG. 2). Gupta further teaches adjusting the rate of execution for each functional unit by removing the power to the functional unit responsive to the power control register field (column 4, lines 5-7). Thus, individual bits (power tokens) are stored and based on their value the power of individual functional units within the microprocessor are turned off. In Gupta, the above-described method allows setting the fields of the power control register by software and allowing better predictability (column 3, lines 41-47). Thus, the hardware task management associated with the processor's power management is reduced and at the same time the power management capabilities are increased (i.e. increased performance) (column 3, lines 47-63).

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to use the method of storing bits indicative of whether certain functional units within the processor are to be turned off, as suggested by Gupta with

the method disclosed by Greene in order to implement the power token comprising a bit to indicate a unit of the processor core to turn off during execution of the instruction. One of ordinary skill in the art would be motivated to do so in order to increase the power management capabilities and performance.

Greene and Gupta fail to disclose predecoding instructions prior to storage of the instructions in a level one cache for a processor core.

DeLano teaches a system and method for predecoding instructions as instructions are copied into cache (column 3, lines 36-39) utilizing a predecode unit (FIG. 1, 108) placed between the memory and the instruction cache (Abstract, lines 3-6, column 5, lines 35-39, FIG. 1, 104, 106). In DeLano, the above-described system and method reduces the time required for deciding whether instructions can be executed simultaneously and provides an indication of which functional unit instructions should be assigned for execution (column 4, lines 35-37). Thus, the instruction processing speed is increased (column 4, lines 8-12, lines 37-39).

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to use the system and method for predecoding instructions prior of storing them into the instruction cache, as suggested by DeLano with the method disclosed by Greene and Gupta in order to implement predecoding instructions prior to storage of the instructions in a level one cache for a processor core. One of ordinary skill in the art would be motivated to do so in order to increase the instruction processing speed and achieve faster power management for the processor core.

Regarding claim 2, Gupta further teaches the method, further comprising adjusting the frequency of the processor core and the voltage of the processor in response to the dynamic power count (column 3, line 64 – column 4, line 10).

Regarding claim 3, Greene further discloses the method, further comprising adjusting the power dissipation in the processor core based upon a threshold associated with the dynamic power count, the threshold being related to a physical limitation of the processor core (column 4, lines 57-62, column 5, line 65 – column 6 line 15, column 7, line 31 – column 8, line 13, column 9, lines 37-48).

Regarding claim 4, DeLano further teaches the method, further comprising modifying execution flags associated with the instruction, the execution flags to schedule issuance of the instruction with respect to other instructions in a parallel execution group that comprises the instruction (column 3, lines 36-51).

Regarding claim 5, Greene further discloses the method, wherein re-encoding the opcode comprises selecting the power token from a look-up table, wherein the look-up table comprises the power token and other power tokens for instructions to be executed by the processor core (column 4, lines 35-50, column 6, lines 51 – column 7, line 16, FIG. 2).

Regarding claim 6, Greene further discloses the method, wherein re-encoding the opcode comprises selecting the new opcode that identifies the instruction and the power dissipation associated with the instruction (column 6, line 51 – column 7, line 30).

Regarding claim 7, DeLano further teaches the method as per claim 6, wherein the opcode and the new opcode utilize an equal number of bits (column 9, lines 65-68, FIG. 3, FIG. 5, 522).

Regarding claim 31, Greene discloses a re-encoder to re-encode an opcode of an instruction to incorporate a power token (column 4, lines 18-23, lines 35-50, column 5, lines 12-21); and

transform control logic to adjust the power dissipation in the processor core based upon management control bits associated with the power dissipation, in response to a dynamic power count for the processor core (column 5, line 12 – column 6, line 15, column 7, line 31 – column 8, line 13, column 9 , lines 37-48, line 66 – column 10, line 15, lines 27-32).

Greene fails to disclose the power token comprising a bit to indicate a unit of the processor core to turn off during execution of the instruction.

Gupta teaches a power control register (FIG. 2, 108) including a plurality of fields for individually controlling the power consumption of the individual functional units within the microprocessor (column 5, lines 45-57, lines 61-67, FIG. 2). Gupta further teaches adjusting the rate of execution for each functional unit by removing the power to the functional unit responsive to the power control register field (column 4, lines 5-7). Thus, individual bits (power tokens) are stored and based on their value the power of individual functional units within the microprocessor are turned off. In Gupta, the above-described method allows setting the fields of the power control register by software and allowing better predictability (column 3, lines 41-47). Thus, the hardware task

management associated with the processor's power management is reduced and at the same time the power management capabilities are increased (i.e. increased performance) (column 3, lines 47-63).

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to use the method of storing bits indicative of whether certain functional units within the processor are to be turned off, as suggested by Gupta with the method disclosed by Greene in order to implement the power token comprising a bit to indicate a unit of the processor core to turn off during execution of the instruction. One of ordinary skill in the art would be motivated to do so in order to increase the power management capabilities and performance.

Greene and Gupta fail to disclose a pre-decoder residing between levels of cache for managing power dissipation in a processor core.

DeLano teaches a system and method for predecoding instructions as instructions are copied into cache (column 3, lines 36-39) utilizing a predecode unit (FIG. 1, 108) placed between the memory and the instruction cache (Abstract, lines 3-6, column 5, lines 35-39, FIG. 1, 104, 106). In DeLano, the above-described system and method reduces the time required for deciding whether instructions can be executed simultaneously and provides an indication of which functional unit instructions should be assigned for execution (column 4, lines 35-37). Thus, the instruction processing speed is increased (column 4, lines 8-12, lines 37-39).

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to use the system and method for predecoding instructions prior of

storing them into the instruction cache, as suggested by DeLano with the method disclosed by Greene and Gupta in order to implement a pre-decoder residing between levels of cache for managing power dissipation in a processor core. One of ordinary skill in the art would be motivated to do so in order to increase the instruction processing speed and achieve faster power management for the processor core.

Regarding claim 32, Gupta further teaches the pre-decoder, further comprising a frequency index coupled with the transform logic to control the frequency of the processor core and a voltage index coupled with the transform control logic to control the voltage of the processor core (column 3, line 64 – column 4, line 10, column 6, line 29 – column 7, line 20, column 7, line 56 – column 8, line 14).

Regarding claim 33, Greene further discloses the pre-decoder, further comprising a threshold buffer having a representation of a physical limitation of the processor core, the physical limitation being related to the dynamic power count, wherein the threshold buffer is coupled with the transform control logic to adjust the power dissipation in the processor core (column 4, lines 35-50, lines 57-62, column 5, line 65 – column 6, line 15, column 7, line 31 – column 8, line 13).

Regarding claim 34, DeLano further teaches the pre-decoder, further comprising an execution flag register coupled with the transform control logic to modify execution flags associated with the instruction, the execution flags to schedule issuance of the instruction with respect to other instructions in a parallel execution group associated with the instruction (column 3, lines 36-51, column 7, lines 37-41, FIG. 3).

Regarding claim 35, Greene further discloses the pre-decoder, wherein the re-encoder comprises a power token table, the power token table having the power token and power tokens for other instructions to be executed by the processor core (column 4, lines 35-50, column 6, lines 51 – column 7, line 16, FIG. 2).

Regarding claim 36, Greene further discloses the pre-decoder, wherein the re-encoder is configured to select a new opcode that identifies other instructions to be executed by the processor core (column 6, line 51 – column 7, line 30, column 10, lines 27-32).

Regarding claim 37, DeLano further teaches the pre-decoder as per claim 36, wherein the opcode and the new opcode utilize an equal number of bits (column 9, lines 65-68, FIG. 3, FIG. 5, 522).

Claims 8, 14, and 18-20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Burns et al. U.S. Patent Appl. Pub. No. 2003/0126479 in view of DeLano et al., U.S. Patent No. 5,337,415.

Regarding claim 8, Burns discloses a method comprising:

monitoring an instruction execution rate for a processor core (paragraph 0021, lines 1-4, paragraph 0026, lines 1-7);

creating a dynamic power count representative of power dissipation in the processor core based upon the instruction execution rate (paragraph 0020, lines 3-6, paragraph 0021, lines 5-16); and

dynamically adjust power dissipation by the processor core based upon the dynamic power count (paragraph 0020, lines 6-9, paragraph 0021, lines 5-19, paragraph 0052, lines 1-8).

Burns fails to disclose per-decoding instructions prior to storage in a level one cache.

DeLano teaches a system and method for predecoding instructions as instructions are copied into cache (column 3, lines 36-39) utilizing a predecode unit (FIG. 1, 108) placed between the memory and the instruction cache (Abstract, lines 3-6, column 5, lines 35-39, FIG. 1, 104, 106). In DeLano, the above-described system and method reduces the time required for deciding whether instructions can be executed simultaneously and provides an indication of which functional unit instructions should be assigned for execution (column 4, lines 35-37). Thus, the instruction processing speed is increased (column 4, lines 8-12, lines 37-39).

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to use the system and method for predecoding instructions prior of storing them into the instruction cache, as suggested by DeLano with the method disclosed by Burns in order to implement per-decoding instructions prior to storage in a level one cache. One of ordinary skill in the art would be motivated to do so in order to increase the instruction processing speed and achieve faster power management for the processor core.

Regarding claim 14, DeLano further teaches the method, wherein pre-decoding comprises identifying a set of the instructions, each instruction of the set having a result that is independent of the results of the other instructions (column 3, lines 36-57).

Regarding claim 18, Burns further discloses the method, wherein pre-decoding comprises incorporating a bit in a power token associated with the instructions to control clock degating of units in the processor core, wherein the units comprise dynamic logic (paragraph 0050, lines 6-11, paragraph 0032, lines 6-10).

Regarding claim 19, Burns further discloses the method, wherein pre-decoding comprises incorporating a bit in a power token associated with the instructions to maintain inputs for units in the processor core, wherein the units comprises static logic (paragraph 0050, lines 6-11, paragraph 0026, lines 1-7, paragraph 0029, lines 1-7, FIG. 2).

Regarding claim 20, Burns further discloses the method, wherein pre-decoding comprises modifying the frequency and voltage associated with the processor based upon the dynamic power count (paragraph 0026, lines 4-7, paragraph 0039, lines 3-5, paragraph 0041, lines 1-10, paragraph 0046, lines 1-13).

Claims 21-26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Burns et al. U.S. Patent Appl. Pub. No. 2003/0126479 in view of Greene, U.S. Patent No. 6,775,787, and further in view of DeLano et al., U.S. Patent No. 5,337,415.

Re claim 21, Burs discloses a method, comprising:

determining a dynamic weighted execution rate associated with instructions executed by the processor core (paragraph 0020, lines 3-6, paragraph 0021, lines 5-16, paragraph 0028, lines 1-10, paragraph 0041, lines 1-3); and

adjusting power dissipation by the processor core based upon the dynamic weighted execution rate (paragraph 0020, lines 3-9, paragraph 0021, lines 5-19, paragraph 0052, lines 1-8).

Burns fails to disclose encoding instructions with a power token, to monitor power dissipation in the processor core.

Greene teaches instruction scheduling based on power estimation (column 1, lines 6-8). Greene further teaches decoding instructions and associating a power value (e.g. bit value from a power value lookup table – FIG. 2, 210) with each instruction to be used by the instruction scheduler, dispatch logic, and power control logic (FIG. 2, 208, 226, and 222) and adjusting the processor's core power state accordingly (column 5, line 65 –column 6, line 15, column 7, line 31- column 8, line 13). Thus, the processor adjusts its power dissipation by monitoring the accumulative power value (column 5, lines 12-64) (created by adding the individual power values associated with each instruction) and scheduling the appropriate instruction execution (i.e. power dissipation) to fall within predetermined thresholds. In Greene, the instruction scheduling provides for adjusting the power consumptions of the processor based not only on hardware resources and data availability, but also accounting for power consumption for instructions that are executing and are to be executed (column 2, lines 4-10). Thus,

halting of the instruction stream is avoided, and performance is increased (column 1, lines 25-26).

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to use the power values associated with each instruction and used for dynamically adjusting the processor's power dissipation as suggested by Greene with the method disclosed by Burns in order to implement encoding instructions with a power token, to monitor power dissipation in the processor core. One of ordinary skill in the art would be motivated to do so in order to increase the performance while adjusting the processor's core power dissipation.

Burns and Greene fail to disclose encoding instructions between levels of cache for a processor core.

DeLano teaches a system and method for predecoding instructions as instructions are copied into cache (column 3, lines 36-39) utilizing a predecode unit (FIG. 1, 108) placed between the memory and the instruction cache (Abstract, lines 3-6, column 5, lines 35-39, FIG. 1, 104, 106). In DeLano, the above-described system and method reduces the time required for deciding whether instructions can be executed simultaneously and provides an indication of which functional unit instructions should be assigned for execution (column 4, lines 35-37). Thus, the instruction processing speed is increased (column 4, lines 8-12, lines 37-39).

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to use the system and method for predecoding instructions prior of storing them into the instruction cache, as suggested by DeLano with the method

disclosed by Burns and Greene in order to implement encoding instructions between levels of cache for a processor core. One of ordinary skill in the art would be motivated to do so in order to increase the instruction processing speed and achieve faster power management for the processor core.

Regarding claim 22, Greene further teaches the method, wherein encoding comprises selecting a power token for the instructions from a table of power tokens, wherein the table comprises a pre-determined power token for each of the instructions (column 6, lines 51-65).

Regarding claim 23, Greene further teaches the method, wherein determining the dynamic weighted execution rate comprises summing the power tokens upon execution of the instruction and adding the sum to the dynamic weighted execution rate (column 5, lines 12-21).

Regarding claim 24, DeLano further teaches the method, wherein adjusting the power dissipation comprises modifying execution flags to adjust an issue rate associated with the instructions (column 3, lines 36-51).

Regarding claim 25, Burns further discloses the method, wherein adjusting the power dissipation comprises adjusting the frequency and voltage of the processor core paragraph 0046, lines 1-13).

Regarding claim 26, DeLano further teaches the method, wherein adjusting the power dissipation comprises adjusting a number of the instructions associated with a group to execute in parallel via the processor core (column 3, lines 36-51, column 4, lines 31-39).

Claims 41 and 45 are rejected under 35 U.S.C. 103(a) as being unpatentable over Greene, U.S. Patent No. 6,775,787 in view of DeLano et al., U.S. Patent No. 5,337,415.

Regarding claim 41, Greene discloses a system comprising:

a summer to sum power tokens associated with instructions executed by a processor core (column 5, lines 30-51, FIG. 2, 250, 252);

an adder coupled with the summer to generate a dynamic weighted execution rate representative of power dissipation in the processor core based upon the sum (column 5, lines 52-60, FIG.2);

a register to maintain a dynamic power count based upon the dynamic execution rate (column 5, lines 60-67, FIG.2); and

a pre-decoder coupled with the register, to associate the power tokens with instructions and to dynamically adjust power dissipation by the processor core based upon the dynamic power count and a state of management control bits (column 5, line 65 –column 6, line 15, column 7, lines 1-16, FIG. 2).

Greene fails to disclose a pre-decoder, residing between main memory and a level one cache for the processor core.

DeLano teaches a system and method for predecoding instructions as instructions are copied into cache (column 3, lines 36-39) utilizing a predecode unit (FIG. 1, 108) placed between the memory and the instruction cache (Abstract, lines 3-6, column 5, lines 35-39, FIG. 1, 104, 106). In DeLano, the above-described system and method reduces the time required for deciding whether instructions can be executed

simultaneously and provides an indication of which functional unit instructions should be assigned for execution (column 4, lines 35-37). Thus, the instruction processing speed is increased (column 4, lines 8-12, lines 37-39).

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to use the system and method for predecoding instructions prior of storing them into the instruction cache, as suggested by DeLano with the method disclosed by Greene in order to implement a pre-decoder, residing between main memory and a level one cache for the processor core. One of ordinary skill in the art would be motivated to do so in order to increase the instruction processing speed and achieve faster power management for the processor core.

Regarding claim 45, Greene further discloses the system, wherein the pre-decoder comprises a base pre-decoder to identify a set of the instructions, each instruction of the set having a result that is independent of the results of the other instructions in the set (column 8, line 50 – column 9, line 20).

Claim 49 is rejected under 35 U.S.C. 103(a) as being unpatentable over Greene, U.S. Patent No. 6,775,787 in view of DeLano et al., U.S. Patent No. 5,337,415, and further in view of Gupta et al., U.S. Patent No. 5,996,083.

Regarding claim 49, Greene and DeLano disclose the system as per claim 41.

Greene and DeLano fail to disclose the pre-decoder is configured to modify a frequency and a voltage associated with the processor core.

Gupta teaches a power control register (FIG. 2, 108) including a plurality of fields for individually controlling the power consumption of the individual functional units within

the microprocessor (column 5, lines 45-57, lines 61-67, FIG. 2). Gupta further teaches modifying the clock signals and operational power for the individual functional units based upon the values stored in the power control register (column 3, line 64 – column 4, line 10). Thus, individual bits (power tokens) are stored and based on their value the clocks and power of individual functional units within the microprocessor are modified. In Gupta, the above-described method allows setting the fields of the power control register by software and allowing better predictability (column 3, lines 41-47). Thus, the hardware task management associated with the processor's power management is reduced and at the same time the power management capabilities are increased (i.e. increased performance) (column 3, lines 47-63).

It would have been obvious to one of ordinary skill in the art at the time of applicant's invention to use the method and system of storing bits indicative of whether the clocks and/or power of certain functional units within the processor are to be adjusted, as suggested by Gupta with the system disclosed by Greene and DeLano in order to implement the pre-decoder is configured to modify a frequency and a voltage associated with the processor core. One of ordinary skill in the art would be motivated to do so in order to increase the power management capabilities and performance.

Allowable Subject Matter

Claims 9-13, 15-17, 27, 42-44, and 46-48 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

The following is a statement of reasons for the indication of allowable subject matter:

Regarding claim 9, the prior art of record fails to disclose or suggest the subject matter of claim 6, "wherein the power token is associated with a relative average power that the instruction consumes during execution".

Regarding claim 15, the prior art of record fails to disclose or suggest the subject matter of claim 8, "wherein pre-decoding comprises delineating a group of the instructions for parallel execution based upon stop bit execution flags associated with the instructions".

Regarding claim 27, the prior art of record fails to disclose or suggest the subject matter of claim 21, "wherein adjusting the power dissipation comprises transmitting a signal to an operating system to switch tasks associated with the processor core".

Regarding claim 42, the prior art of record fails to disclose or suggest the subject matter of claim 41, "wherein each power token indicates a relative average power that the associated instruction of the instructions is to consume during execution by the processor core".

Regarding claim 46, the prior art of record fails to disclose or suggest the subject matter of claim 41, "wherein pre-decoding comprises a base pre-decoder to delineate a group of instructions for parallel execution based upon stop bit execution flags associated with the instructions".

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Stefan Stoynov whose telephone number is (571) 272-4236. The examiner can normally be reached on 8:00AM-4:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Lynne Browne can be reached on (571) 272-3670. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

SS